1. In a computing device that may be communicatively coupled to a network by a communication path that includes one or more protocol stacks associated with an abstract interface for managing a filter stack, the filter stack including one or more filter instances, which may perform filtering operations on data packets being transferred via the communication path, a method for inserting a filter instance into the filter stack without disrupting the operation of associated protocol stacks so as to conserve processor and network resources, the method comprising:

an act of pausing operation of the filter stack;

an act of inserting the filter instance into the filter stack while at least one associated protocol stack continues to be capable of transferring data; and

an act of starting operation of the filter stack.

2. The method as recited in claim 1, wherein pausing operation of the filter stack comprises the following:

an act of pausing one or more filter instances included in the filter stack.

3. The method as recited in claim 2, wherein pausing one or more filter instances included in the filter stack comprises the following:

an act of one or more pause routines receiving data indicating that the one or more filter instances should be paused.

4. The method as recited in claim 1, wherein pausing operation of a filter stack comprises the following:

an act of redirecting a transferred data packet to a dummy routine that returns the data packets to the communication path without modifying the data included in the data packet.

5. The method as recited in claim 1, wherein inserting the filter instance into the filter stack comprises the following:

an act of inserting a filter instance that was configured by using parameters received from the abstract interface.

6. The method as recited in claim 1, wherein inserting the filter instance into the filter stack comprises the following:

an act of a filter driver receiving a filter handle that may be used to facilitate transferring data to an abstract interface.

7. The method as recited in claim 1, wherein inserting the filter instance into the filter stack comprises the following:

an act of a filter driver allocating resources for the filter instance.

8. The method as recited in claim 1, wherein inserting the filter instance into the filter stack comprises the following:

an act of a filter driver creating a filter instance context for the filter instance.

9. The method as recited in claim 8, wherein a filter driver creating a filter instance context for the filter instance comprises the following:

an act of the filter driver sending the filter instance context to the abstract interface.

10. The method as recited in claim 1, wherein inserting the filter instance into the filter stack comprises the following:

an act of a filter driver registering data with the abstract interface.

11. The method as recited in claim 1, wherein inserting the filter instance into the filter stack comprises the following:

an act of a filter driver registering data in a system registry.

12. The method as recited in claim 1, wherein inserting the filter instance into the filter stack comprises the following:

an act of inserting a filter instance that was configured by using parameters received from a system registry.

13. The method as recited in claim 1, wherein inserting the filter instance into the filter stack comprises the following:

an act of inserting the filter instance in a predetermined location in the filter stack.

14. The method as recited in claim 1, wherein inserting the filter instance into the filter stack comprises the following:

an act of inserting a filter instance that is capable of filtering data packets transferred over virtual connections.

15. The method as recited in claim 1, wherein inserting the filter instance into the filter stack comprises the following:

an act of inserting a filter instance that includes an entry point to receive data associated with the power management of the computing device.

16. The method as recited in claim 1, wherein inserting the filter instance into the filter stack comprises the following:

an act of inserting a filter instance that includes an entry point to receive data associated with plug and play devices.

17. The method as recited in claim 1, wherein inserting the filter instance into the filter stack comprises the following:

an act of inserting a filter instance that includes properties that may be modified through a management interface.

18. The method as recited in claim 1, wherein inserting the filter instance into the filter stack comprises the following:

an act of inserting a filter instance on a filter stack that is bound to a plurality of transport layer protocols.

19. The method as recited in claim 1, wherein inserting the filter instance into the filter stack comprises the following:

an act of a filter driver verifying that the filter instance was inserted into the filter stack.

20. The method as recited in claim 19, wherein a filter driver verifying that the filter instance was inserted into the filter stack comprises the following:

an act of the filter driver sending an insertion status to the abstract interface.

21. The method as recited in claim 1, wherein starting operation of the filter stack comprises the following:

an act of starting one or more filter instances included in the filter stack.

22. The method as recited in claim 21, wherein starting one or more filter instances included in the filter stack comprises the following:

an act of one or more start routines receiving data indicating that the one or more filter instances should be started.

23. The method as recited in claim 1, further comprising:

an act of notifying associated protocol stacks that operation of the filter stack is going to be paused.

24. The method as recited in claim1, further comprising:

an act of notifying associated protocol stacks that the filter stack now includes the inserted filter instance.

25. The method as recited in claim 1, further comprising

an act of notifying associated protocol stacks that operation of the filter stack is going to be started.

26. In a computing device that may be communicatively coupled to a network by a communication path that includes one or more protocol stacks associated with an abstract interface for managing a filter stack, the filter stack including one or more filter instances, , a method for removing a filter instance from the filter stack without disrupting the operation of associated protocol stacks so as to conserve processor and network resources, the method comprising:

an act of pausing operation of the filter stack;

an act of removing the filter instance from the filter stack while at least one associated protocol stack continues to be capable of transferring data; and

an act of starting operation of the filter stack.

27. The method as recited in claim 26, wherein pausing operation of the filter stack comprises the following:

an act of pausing one or more filter instances included in the filter stack.

28. The method as recited in claim 27, wherein pausing one or more filter instances included in the filter stack comprises the following:

an act of one or more pause routines receiving data indicating that the one or more filter instances should be paused.

29. The method as recited in claim 26, wherein pausing operation of a filter stack comprises the following:

an act of redirecting transferred data packets to dummy routines that return the data packets to the communication path.

30. The method as recited in claim 26, wherein removing the filter instance from the filter stack comprises the following:

an act of pausing the filter instance.

31. The method as recited in claim 26, wherein removing the filter instance from the filter stack comprises the following:

an act of a filter driver releasing resources associated with the filter instance.

32. The method as recited in claim 26, wherein starting operation of the filter stack comprises the following:

an act of starting one or more filter instances included in the filter stack.

33. The method as recited in claim 32, wherein starting one or more filter instances included in the filter stack comprises the following:

an act of one or more start routines receiving data indicating that the one or more filter instances should be started.

34. The method as recited in claim 26, further comprising:

an act of notifying associated protocol stacks that operation of the filter stack is going to be paused.

35. The method as recited in claim 26, further comprising:

an act of notifying associated protocol stacks that filter instance has been removed from the filter stack.

36. The method as recited in claim 26, further comprising

an act of notifying associated protocol stacks that operation of the filter stack is going to be started.

37. In a computing device that may be communicatively coupled to a network by a communication path that includes one or more protocol stacks associated with an abstract interface for managing a filter stack, which may perform filtering operations on portions of data packets being transferred via the communication path, a method for inserting a filter instance into the filter stack without disrupting the operation of associated protocol stacks so as to conserve processor and network resources, the method comprising:

an act of pausing operation of the filter stack; and

a step for reconfiguring filtering operations included in the filter stack in a manner that promotes efficient transfer of data along the communication path during reconfiguration.

38. In a computing device that may be communicatively coupled to a network by one or more communication paths and that includes one or more protocol stacks associated with an abstract interface that manages one or more filter instances that may perform filtering operations on packets, a method for filtering packets associated with the one or more protocols so as to conserve processor and network resources, the method comprising:

an act of processing an input packet;

an act of the input packet bypassing one or more of the filtering operations associated with a filter instance; and

an act of sending an output packet that may differ from the input packet if a filtering operation was performed.

39. The method as recited in claim 38, wherein processing an input packet comprises the following:

an act of processing an input packet associated with a data link layer.

40. The method as recited in claim 38, wherein processing an input packet associated comprises the following:

an act of processing an input packet associated with a data link layer driver.

41. The method as recited in claim 38, wherein processing an input packet comprises the following:

an act of processing an input packet associated with a transport layer protocol.

42. The method as recited in claim 41, wherein processing an input packet associated with a transport layer protocol comprises the following:

an act of processing an input data packet associated with the Transmission Control Protocol.

43. The method as recited in claim 41, wherein processing an input packet associated with a transport layer protocol comprises the following:

an act of processing an input data packet associated with the Internet Protocol.

44. The method as recited in claim 41, wherein processing an input packet associated with a transport layer protocol comprises the following:

an act of processing an input packet from a transport layer driver.

45. The method as recited in claim 38, wherein processing an input packet compromises the following:

an act of processing a data packet transferred along a data path from a transport layer to a data link layer.

46. The method as recited in claim 38, wherein processing an input packet compromises the following:

an act of processing a data packet transferred along a data path from a data link layer to a transport layer.

47. The method as recited in claim 38, wherein processing an input packet compromises the following:

an act of processing a control packet, which includes request information, and is being transferred along a control path.

48. The method as recited in claim 38, wherein processing an input packet compromises the following:

an act of processing a control packet, which includes system indication information, and is being transferred along a control path.

49. The method as recited in claim 38, wherein the input packet bypassing one or more of the filtering operations associated with a filter instance comprises the following:

an act of a received input packet bypassing one or more internal filtering operations associated with a filter instance included in a filter stack.

50. The method as recited in claim 38, wherein the input packet bypassing one or more of the filtering operations associated with a filter instance comprises the following:

an act of a received input packet bypassing one or more internal filtering operations associated with a filter instance based on characteristics associated with the input packet.

51. The method as recited in claim 38 wherein, the input packet bypassing one or more of the filtering operations associated with a filter instance comprises the following:

an act of the input packet completely bypassing the filter instance.

52. The method as recited in claim 51, wherein the input packet completely bypassing the filter instance comprises the following:

an act of the input packet completely bypassing the filter instance because the filter instance was configured to operate in bypass mode for the communication path along which the input packet is being transferred.

53. The method as recited in claim 50, wherein the input packet bypassing one or more filtering operations associated with a filter instance based on characteristics associated with the input packet comprises the following:

an act of bypassing a first set of filtering operations when an input packet is associated with a first protocol and bypassing a second set of filtering operations when an input packet associated with a second protocol.

54. The method as recited in claim 53, wherein the first set of filtering operations and the second set of filtering operations are different.

55. The method as recited in claim 38, wherein sending an output packet that may differ from the input packet if a filtering operation was performed on the input packet comprises the following:

an act of sending an output packet that may differ from the input packet if a filtering operation modified the input packet.

56. In a computing device that may be communicatively coupled to a network by one or more communication paths and that includes one or more protocol stacks associated with an abstract interface that manages a filter stack, a method for filtering packets associated with the one or more protocols so as to conserve processor and network resources, the method comprising:

an act of receiving an input packet associated with a protocol;

a step for transferring an output packet the may have been filtered by some but not all of the filtering operations included in a filter stack so as to increase efficiency of the filtering process.

57. In a computing device that may be communicatively coupled to a network by one or more communication paths and that includes one or more protocol stacks associated one or more filter instances that may perform filtering operations on packets, a method for filtering packets associated with the one or more protocols so as to conserve processor and network resources, the method comprising:

an act of a filter instance receiving an input packet;

an act of bypassing a first set of filtering operations when the input packet is transferred across a communication path from a data link layer to a transport layer and bypassing a second set of filtering operations when the input packet is transferred across a communication path from a transport layer to a data link layer; and

an act of sending an output packet that may differ from the input packet if a filtering operation was performed.

58. The method as recited in claim 57, wherein the first set of filtering operations and the second set of filtering operations are different.

59. The method as recited in claim 57, wherein the communication path is a data path.

60. The method as recited in claim 57, wherein the communication path is a control path.

61. In a computing device that may be communicatively coupled to a network by one or more communication paths and that includes one or more protocol stacks associated one or more filter instances that may perform filtering operations on packets, a method for filtering packets associated with the one or more protocols so as to conserve processor and network resources, the method comprising:

an act of a processing a packet currently being transferred across a communication path;

an act of the packet completely bypassing a first filter instance in the communication path because the first filter instance was configured to operate in bypass mode for the communication path; and

an act of the packet being received by a second filter instance in the communication path that was configured to receive packets being transferred across the communication path.

62. The method as recited in claim 61, wherein the packet completely bypassing a first filter instance in the communication path because the first filter instance was configured to operate in bypass mode for the communication path comprises the following:

an act of the packet completely bypassing a first filter instance in the communication path because the first filter instance was configured by an abstract interface to operate in bypass mode for the communication path

63. The method as recited in claim 62, further comprising:

an act of the abstract interface configuring the filter instance to resume receiving packets.

64. The method as recited in claim 61, an act of the packet completely bypassing a first filter instance in the communication path because the first filter instance was configured to operate in bypass mode for the communication path wherein comprises the following:

an act of an input data packet completely bypassing a first filter instance in a data path because the first filter instance was configured to operate in bypass mode for the data path.

65. The method as recited in claim 61, an act of the packet completely bypassing a first filter instance in the communication path because the first filter instance was configured to operate in bypass mode for the communication path wherein comprises the following:

an act of an input control packet completely bypassing a first filter instance in a control path because the first filter instance was configured to operate in bypass mode for the control path.

66. A computer program product for implementing, in a computing device that may be communicatively coupled to a network by a communication path that includes one or more protocol stacks associated with an abstract interface for managing a filter stack, which may perform filtering operations on portions of data packets being transferred via the communication path, a method for inserting a filter instance into the filter stack without disrupting the operation of associated protocol stacks so as to conserve processor and network resources , the computer program product comprising:

a computer-readable medium carrying computer-executable instructions, that when executed at the computing device cause the computing device to perform the method, including:

an act of pausing operation of the filter stack,

an act of inserting the filter instance into the filter stack while at least one associated protocol stack continues to be capable of transferring data; and

an act of starting operation of the filter stack.

67. A computer program product for implementing, in a computing device that may be communicatively coupled to a network by a communication path that includes one or more protocol stacks associated with an abstract interface for managing a filter stack, the filter stack including one or more filter instances, which may perform filtering operations on portions of data packets being transferred via the communication path, a method for inserting a filter instance into the filter stack without disrupting the operation of associated protocol stacks so as to conserve processor and network resources, the computer program product comprising:

a computer-readable medium carrying computer-executable instructions, that when executed at the computing device cause the computing device to perform the method, including:

an act of pausing operation of the filter stack;

an act of removing the filter instance from the filter stack while at least one associated protocol stack continues to be capable of transferring data; and

an act of starting operation of the filter stack.

68. A computer program product for implementing, in a computing device that may be communicatively coupled to a network by one or more communication paths and that includes one or more protocol stacks associated with an abstract interface that manages a filter stack, the filter stack including one or more filter instances that may perform filtering operations on portions of packets, a method for filtering packets associated with the one or more protocols so as to conserve processor and network resources, the computer program product comprising:

a computer-readable medium carrying computer-executable instructions, that when executing at the computing device cause the computing device to perform the method, including:

an act of processing an input packet;

an act of the input packet bypassing one or more of the filtering operations associated with a filter instance; and

an act of sending an output packet that may differ from the input packet if a filtering operation was performed on the input packet.